

## Ass2

### Testing

```
make -s run_test
```

### Using the client

```
make build_client  
cd client
```

```
# Try a command!  
./WTF create test_project
```

### Using the server

```
make build_server  
cd server
```

```
# Start the server  
./WTFServer 8000
```

Or, use the shortcut.

```
# Start the server (port 8000 by default)  
make run_server
```

### Storing files in memory

The only files read are those listed in `.Manifest`. To send files, read in the manifest, then read file data into every file listed in the manifest.

```
// Read in the manifest  
Manifest* manifest = manifest_read("projects/test_project");  
  
// Get files listed in the manifest  
// Each file contains `file_name`, `file_version` and `file_hash`. See `filelist.h`  
FileList* manifest_files = manifest->filelist;  
  
// Read file data into every listed file  
// Each file now now also contains `file_size` and `file_bytes`. See `filelist.h`  
FileList* files_with_data = filelist_readbytes("projects/test_project", manifest_files);
```

### Sending files

After successfully storing files inside of a `FileList` linked list, add them to a request to send them. Open `request.h` to see what else that a request can contain.

```

// Read in manifest
Manifest* manifest = manifest_read("projects/test_project");

// Get files listed in the manifest
FileList* manifest_files = manifest->filelist;

// Read file data into every listed file
FileList* files_with_data = filelist_readbytes("projects/test_project", manifest_files);

// Create request
Request* request = request_new();

// Send files with request
request->filelist = files_with_data;

Response* response = client_send(request);

```

## Installing OpenSSL

Taken from here.

```

brew update
brew install openssl
echo 'export PATH="/usr/local/opt/openssl/bin:$PATH"' >> ~/.bash_profile
source ~/.bash_profile

```

## Protocol

### Response Protocol

```

<message>:
<status_code>:

<project_version>:
<file_count>:
<file_path>:
<file_version>:
<file_hash>:
<file_size>:
<file_removed>:
<file_bytes>

```

### Request Protocol

```

<message>:
<status_code>:

```

```
<command_name>:  
<project_name>:  
  
<project_version>:  
<file_count>:  
<file_path>:  
<file_version>:  
<file_hash>:  
<file_size>:  
<file_removed>:  
<file_bytes>:
```